Radio Environment Maps as a Tool for Occupancy Detection in Indoor Environment

Toufiq Aziz[1], Shafi Ullah Khan[2] and Insoo Koo[3]

[1,3]University of Ulsan, Ulsan 44610, The Republic of Korea.
[2]University of Texas at Arlington, United States.
toufiq97@mail.ulsan.ac.kr[1], shafiullah.khan@uta.edu[2] and iskoo@ulsan.ac.kr[3]

Abstract. Smart building functionalities are crucial for optimizing energy consumption through accurate occupancy detection in indoor environments. This study explores a novel approach to occupancy detection by leveraging Radio Environment Maps (REMs). Unlike conventional methods, the proposed technique involves creating detailed REMs across diverse indoor environments, considering variables such as room size, furniture arrangements, and the presence of the occupants. A predictive model is then applied to the REM data to detect occupancy and validate the accuracy of the predictions. The method was tested across multiple scenarios, including rooms with and without occupants, demonstrating its adaptability to different environmental conditions. By analyzing signal behavior variations, this method provides insights into how environmental factors impact detection performance. The results demonstrate the robustness of the proposed REM-based approach, achieving high prediction accuracy and confirming the effectiveness of the REMs in capturing the environmental nuances that influence signal behavior. This approach offers a flexible and scalable alternative for occupancy detection, with potential applications in energy-efficient building management and smart environment design.

Key Words: Building energy saving (BES), Radio Environment map (REM), Turtlebot3 (Robot), Access Point (AP), Occupancy Monitoring, Machine Learning.

1. Introduction

Through precise occupancy detection, technological advancements are improving occupant comfort and lowering energy usage in smart buildings. Accurate information about the presence of the occupants enables effective control of lighting, HVAC, and ventilation systems, reducing energy consumption without sacrificing comfort [1]. Existing occupancy detection techniques, however, frequently encounter difficulties such as exorbitant expenses, upkeep needs, restricted scalability, and privacy issues. These limitations highlight the need for more advanced and cost-effective solutions to further improve energy, efficiency and occupant comfort in smart buildings [1, 2].

These drawbacks emphasize the need for more sophisticated and flexible occupancy detection systems, and new technologies are being investigated to get over these obstacles. The use of Wireless Sensor Networks (WSNs), which entail placing networks of low-power sensors throughout a building to gather information on different environmental characteristics, is one such strategy [3-5]. From there, occupancy patterns can be deduced. Furthermore, without the need for extra hardware, machine learning and artificial intelligence algorithms are being used to evaluate data from many sources, including the infrastructure already present in buildings, to improve occupancy detection accuracy. Additionally, thermal imaging is used in infrared thermography to identify stationary people more accurately by detecting their presence based on body heat. Radio-frequency (RF) sensing is another non-intrusive technique that uses RF signals, such as Wi-Fi or specialized RF devices, to identify changes in the environment brought on by human movement. Additionally, by integrating data from smartphones and wearable technology, like smart badges or cell phones, occupancy levels can be calculated while privacy issues are addressed through opt-in procedures [6,7].

Moreover, substantial energy savings, better occupant experiences, and increased safety and security can result from integrating these cutting-edge occupancy detection systems with building management systems (BMS) [8]. By modifying ventilation rates according to actual occupancy, real-time occupancy data allows for dynamic control of HVAC systems, lighting, and other utilities, improving comfort and lowering energy waste while

improving indoor air quality. Accurate occupancy detection enhances safety and security by helping first responders know the precise number and location of inhabitants in an emergency, improving evacuation protocols [8]. In summary, while conventional occupancy detection techniques have given smart buildings their fundamental capabilities, their drawbacks underscore the need for more sophisticated, scalable, and privacy-aware alternatives. Ongoing research and innovation in this domain are crucial to fully harness the promise of intelligent building systems, hence fostering more sustainable and occupant-friendly settings. Integrating radio environment maps (REMs) with the BMS unlocks a new realm of possibilities [4,5]. The BMS is able to make modifications instantly through the real-time occupancy data obtained from the received signal strength indicator (RSSI) signals. Lighting systems become responsive, illuminating paths as individuals move and dimming in unoccupied areas. Ventilation can adapt to maintain optimal air quality, directly responding to the actual number of people in a room rather than relying on fixed schedules or inaccurate sensors [9].

Acknowledging these constraints, the pursuit of increasingly sophisticated and flexible solutions has accelerated. Making radio environment maps by using RSSI signals is a great approach in wireless networks. Wireless signals, ranging from Bluetooth to Wi-Fi, are ubiquitous throughout contemporary architecture. People unintentionally interact with these signals when they pass through these areas. Bodies absorb and reflect radio waves, causing minute variations in signal strength [9-11].

By meticulously measuring these RSSI variations, it's possible to construct a dynamic map of the indoor radio environment. This REM becomes a living blueprint of occupancy and movement within the building. Advanced algorithms analyze the flow of signal strengths, discerning patterns that correlate with the presence and number of occupants all without the need for invasive sensors or cameras [12-15]. Beyond comfort and energy efficiency, accurate occupancy detection enhances safety and security. In emergency situations such as fires or evacuations first responders can save lives by using precise information about where people are located within a building, allowing them to focus efforts where they're needed most. Recent advancements for occupancy detection in smart buildings have introduced innovative methods to enhance energy efficiency and functionality. Notable contributions include IoT-based sensor fusion approach, audio-processing techniques for non-intrusive estimation, and comprehensive review highlighting the strengths and limitations of current methods [16-18].

The journey toward smarter buildings is not just about technology but also about respecting the occupants' privacy and ensuring scalability. Utilizing RSSI-based REMs addresses these concerns. Since the system relies on ambient radio signals and doesn't capture personal data or visual images, it offers a non-intrusive solution that can be scaled across large and complex facilities without significant additional infrastructure [3, 4].

While traditional methods have laid the foundation for smart buildings, they struggle to adapt to modern, dynamic environments [19, 20]. A promising approach involves using REMs with RSSI signals. This method combines existing technologies with advanced analytics to create spaces that are not only intelligent but also prioritize privacy and resource efficiency. As ongoing research and development in this field continue, we are poised to build structures that are truly responsive to human needs, marking a new era of sustainable and human-centric design [4].

Our goal is to create an accurate REM to determine whether an area is occupied. For precise occupancy detection, the REM relies heavily on the RSSI signal from the Access Point (AP). A Turtlebot3 Burger robot collected real data within room 502 of the Department of Electronics at the University of Ulsan, South Korea. This data has been employed in a well-trained machine learning algorithm to ensure the REM construction is accurate. The REM is then utilized to detect occupancy.

Highlights of our methodology as mentioned below:

We present an occupancy detection method using a machine learning algorithm that constructs a REM and analyzes signal variations in occupied and unoccupied scenarios utilizing RSSI signals.

We constructed a REM using machine learning algorithms that process RSSI values collected by a mobile robot. A Wi-Fi router served as an access point. This technique has proven effective for visualizing the REM. Our study focuses on detecting occupancy within the coverage map by utilizing RSSI signals in an indoor environment.

We looked closely at different popular machine learning methods, like extra tree regressor, random forest regression and bagging regressor. For error calculation we consider Root Mean Square Error (RMSE) and Mean Absolute Error (MAE). Our goal is to find the best regression method for our approach while considering occupancy detection errors.

2. Measurement Strategy

This research focuses on the development and application of a REM for occupancy detection. For our experiments, we are utilizing an indoor environment: a classroom in the building-7 at the University of Ulsan, the republic of Korea. Two devices were used for data collection: the Turtlebot3 Burger and the Hovermap. The TurtleBot3 Burger, a mobile robot outfitted with sensors and electronics, was at the heart of our data collection Procedure. The robot is equipped with several primary components. A Raspberry Pi single-board computer is used for processing. An OpenCR embedded controller is included for control functions. A 180-degree LDS-02 LiDAR sensor is used for mapping and navigation. An inertial measurement unit, along with an encoder, was incorporated into the robot. All components are connected and managed through the Robot Operating System (ROS) to ensure efficient communication [4, 5, 23].

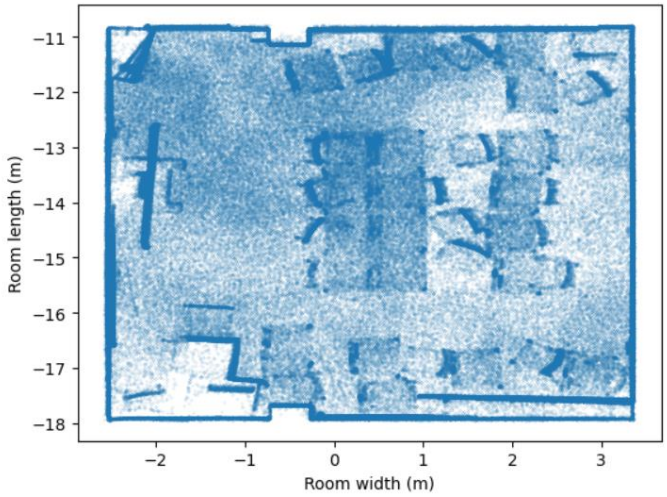


Figure 1. 2D map of the experimental room.

The TurtleBot3 explored the indoor area independently, collecting data in a systematic manner to aid in the creation of a comprehensive REM. We collected a total of '9,000' data points within an area measuring 7 by 6 meters with a height of 2.5 meters. The environment is shown in Figure 1. In addition to the TurtleBot3, we used the Emesent Hovermap where a LiDAR 3D scanner that improved data collecting by offering detailed two-dimensional environmental mapping and SLAM algorithms. Data synchronization was methodically carried out utilizing timestamps to accurately synchronize the RSSI and localization information. An IPTime router acted as the AP signal source operating at 2.4 GHz. The router provided steady signals to support accurate RSSI measurements. The TurtleBot3's built-in Wi-Fi module collected the RSSI data. The iwconfig utility captured network interface information, including the SSID, signal frequency, and link quality [4, 23].

A thorough study was conducted using several regression techniques. This study involved merging the complete dataset with the REM-based occupancy detection algorithm. The dataset included the robot's 2D dimension values along with RSSI values. The Extra Trees Regressor (ETR) algorithm was found to be particularly effective. Continuous target variables were predicted more accurately using this algorithm. This improved the accuracy of the coverage map constructed based on RSSI signals [5].

The TurtleBot3's ability to acquire precise data was demonstrated through this technique. The potential of enhanced REM and occupancy detection systems was also highlighted. These systems were shown to significantly improve human understanding and interaction with intricate interior environments. These approaches are being

used in this research to advance the accuracy of occupancy detection. This development opens the door for more efficient utilization in automated monitoring systems, energy efficiency, and smart building supervision.

3. Experimental Setup

In our suggested method, we created an indoor REM for occupancy detection. The dimension values were initially used as features in the REM construction. The signal values were used as labels. A variety of regression algorithms were applied to the dataset. Among these, the Extra Trees Regressor was found to outperform other methods. The primary focus of this paper is on leveraging RSSI values from the REM, which are analysed using regression techniques, for accurate occupancy identification. Extra Trees Regression is a supervised learning approach that uses a labelled dataset with input characteristics and target values to train. ETR predicts a continuous target variable using input characteristics. This method is particularly suited for solving complex regression problems. It has been found to be highly effective in such scenarios [3, 4].

In the system model, RSSI signals are used to create a coverage map. This coverage map is then utilized to develop the introduced Machine Learning Model for occupancy detection. RSSI is recognized as a critical statistic in wireless communication. It provides valuable insights into signal quality. It is also applied in network design, signal mapping, and device localization. Our work is largely focused on using Wi-Fi technology to collect RSSI data.

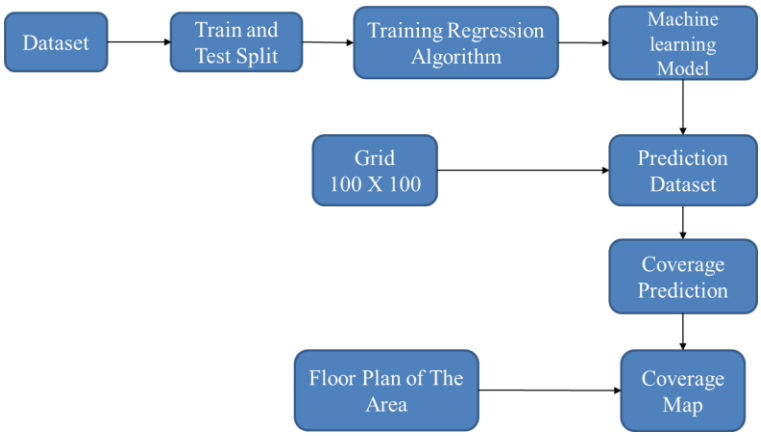


Figure 2. Radio Environment map Construction.

The entire dataset was collected by a mobile robot. It was made sure to incorporate a range of data kinds. The dataset was chosen in order to facilitate occupancy detection and satisfy certain conditions for developing the indoor REM. The data was then preprocessed using train-test splitting and normalization methods. For building the coverage map, the values on the dimension were identified as features, which are x and y, while the RSSI value is a label. Figure 2 shows the usage of a machine learning a method to predict RSSI values. In addition, we created a grid with a consistent spacing of 100 units for the visual graphical depiction. We then tested several regression algorithms to construct the REM, using sophisticated methodologies to assure accurate modelling. We assessed the effectiveness of various methods using RMSE and MAE values to determine the best regressor for our suggested technique while accounting for occupancy detection error.

4. ETR Framework for REM

The Extra Trees algorithm consists of several independent decision trees, each of which uses the whole training dataset. A decision tree is designed as a hierarchical structure consisting of root, child, and leaf nodes. One of the decision trees from an Extra Trees Regressor is shown in Figure 3. At the root node, the algorithm is initiated. Random split points and randomized subsets of features are used to progressively create split procedures. At each child node, this process is repeated. Until a leaf node is encountered, the route keeps going. The decision-making process is embedded inside the structure of the tree [21, 22].

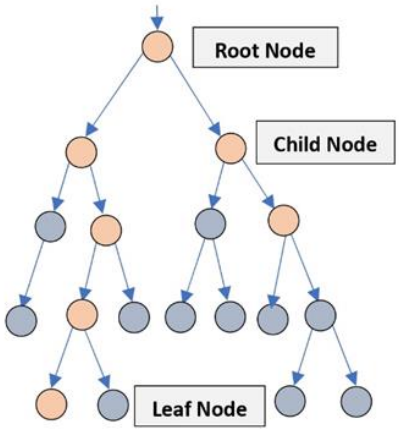


Figure 3. One of the Decision Trees from an Extra Trees Regressor.

In our present study, we used the Extra Trees Regressor to generate indoor REMs for occupancy identification. Using RSSI values as features and labels, we trained the model to predict signal strength fluctuations caused by varying occupancy levels. We assessed our model's performance using error measures such as RMSE, and MAE values. These criteria enabled us to evaluate the Extra Trees Regressor's accuracy in detecting occupancy, guaranteeing that our methodology overcomes the limits of existing approaches while also meeting the objectives of smart building applications [23]. Several supervised learning approaches are available. Tree-based ensemble methods, such as Random Forests and Extra Trees, are commonly used. These methods have gained prominence for addressing classification and regression problems. These strategies use the combined power of many models to improve performance by generating independent base learners using alternative training algorithms or including randomization into the tree construction process. Randomization approaches provide higher variation among trees, resulting in lower correlation and increased independence. However, creating ensembles may be computationally challenging, particularly when working with huge datasets. To address this issue, the attention has turned to the speedier Extra Trees algorithm.

The ETR algorithm creates M autonomous decision trees using a training dataset $Z=\{z_1,z_2,...,z_n\}$ with n ranging from 1 to N samples and z_N being a $(L+2)$- dimensional vector with L features and two target values to forecast. Within each tree, each child node r is allocated the training subset D_s, obtained from dataset X. The ETR method generates the best splitting rule at each node r, using the training subset D_s and a randomly selected subset of H features (where $H \leq L$), as explained in Algorithm 1.

Algorithm 1: Split Rule Optimization in ETR Scheme

1. **Input:** Training subset $D_s = \{s_1, s_2, ..., s_{Ws}\}$
   L-dimensional vector derived from sample $s_k = \{g_1, g_2, ..., g_L\}$
   H = The number of randomly chosen traits
   p_{min} = The bare minimum of samples needed to complete a split
2. **If** $W_s < p_{min}$ or the node contains a label for each observation:
   Stop splitting after designating the node as a leaf node.
3. **Else**:
   Select a random subset of H features $\{g_1, g_2, ..., g_H\}$ from the original L features. $(H \leq L)$
   4. **For** each feature h in the subset, **Do**:
   Determine g_h^{min} and g_h^{max} as the minimum and maximum values of feature h within subset D_s.
   Generate a random split value, g_h^c, uniformly distributed within the range $[g_h^{min}, g_h^{max}]$.
   Set $[g_h < g_h^c]$ as a candidate split.
   End for

5. Select a split $[g_* < g_*^c]$ such that:
   $\text{MSE}(g_*^c) = \min_{\{h=1,2,\ldots,H\}} \text{MSE}(g_h^c)$
6. **Output:** Best split rule $[g_* < g_*^c]$ at child node s.

D_s^{right}, which contains the samples satisfying the splitting condition, and D_s^{left}, which includes the remaining samples are two subsets inside the subset Ds at child node r, specifically. Mean Squared Error is used as the scoring function to help choose the best split. This splitting process is repeated at each child node until the number of samples in D_s reaches the minimum threshold n_{min} required for splitting or until all samples in D_s share the same label. The label assigned to the samples in D_s defines the output of the leaf node. A test sample moves through each decision tree during the testing phase, passing through child nodes according to the best splits, which leads it to the left or right child node. Until the test sample reaches a leaf node, this traversal keeps going.

5. Performance Evaluation

A proper explanation of the error metrics is provided in this section. These metrics are used to evaluate the construction of the REM. As mentioned in the previous section, the dataset was gathered by the turtlebot. This dataset is used to build the REM. The features in the dataset represent positions with x and y coordinates, measured in meters. The target variable is the RSSI, which is measured in dBm.

One of the key metrics for evaluating the average error of the predictions is the MAE. Here, the actual RSSI value is denoted as y_i , and the predicted RSSI value, generated using regression techniques, is represented as $\hat{y}_i$. To assess the accuracy of the predictions, MAE calculates the average of the absolute differences between the true RSSI value y_i and their corresponding predicted values $\hat{y}_i$ [4]. Then, MAE can be expressed by:

$$MAE = \frac{1}{n_{samples}} \sum_{n}^{i=0} |y_i - \hat{y}_i| \tag{1}$$

The RMSE is a metric used to measure the error rate by calculating the square root of the mean of the squared differences between the predicted electric field strength, denoted as $\hat{y}_i$, and the actual observed value, represented as y_i This metric provides insight into how closely the predictions align with the true values. The RMSE can be expressed as:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (\hat{y}_i - y_i)^2}{n}} \tag{2}$$

Table 1 and Table 2 summarize the performance of three machine learning regression models—Extra Tree Regression, Random Forest Regression (RFR), and Bagging Regression (BR) evaluated using RMSE and MAE under two different scenarios: general regression performance and REM error calculations without occupancy.

In Table 1, Extra Tree Regression demonstrates superior accuracy compared to the other models, achieving the lowest error rates in both RMSE and MAE. RFR follows closely, while BR exhibits significantly higher errors, making it the least accurate among the three. Similarly, in Table 2, ETR outperforms the other models in REM error calculations without occupancy, maintaining the lowest error rates. RFR shows moderately higher errors, while BR again records the highest error values. These results collectively highlight the consistent and superior performance of ETR across different scenarios, making it the most reliable model for the given tasks.

Table 1. Performance between the regression techniques based on REM error calculations with occupancy.

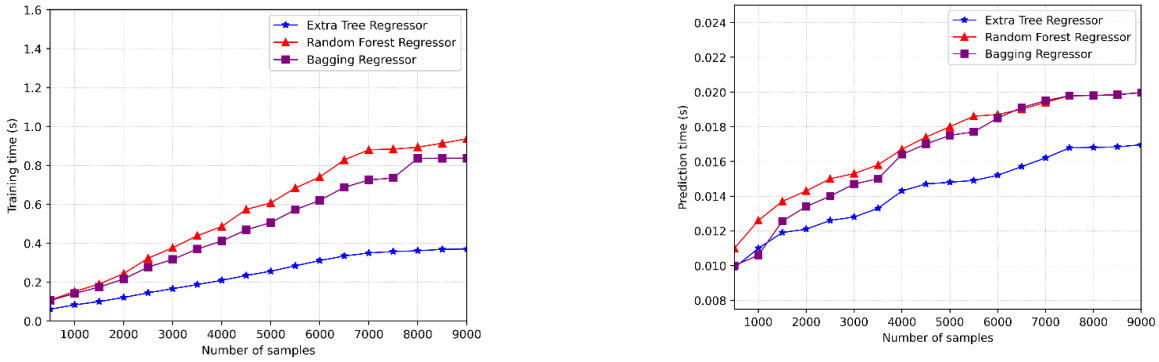
Machine Learning Algorithm	RMSE	MAE
Extra Tree Regression	1.001	0.469
Random Forest Regression	1.09	0.58

Bagging Regression	1.28	0.551

Table 2. Performance between the regression techniques based on REM error calculations without occupancy.

Machine Learning Algorithm	RMSE	MAE
Extra Tree Regression	0.98	0.451
Random Forest Regression	1.02	0.492
Bagging Regression	1.19	0.501

Figure 4 illustrates two graphical representations comparing the computational times of three regression models—ETR, EFE, and BR—as the number of samples increases from 1,000 to 9,000. The first graph (a) shows the training time, where the ETR consistently requires the least time, followed by the RFR and then the BR, which takes the longest. The second graph (b) details the prediction times, displaying a similar trend where the ETR is the most efficient, indicating a faster processing capability both in training and prediction phases across varying sample sizes. These results suggest that the ETR offers superior computational efficiency, making it an advantageous choice for handling large datasets in scenarios where time efficiency is crucial.



(a) Training time vs. Samples

(b) Prediction time vs. Samples

Figure 4. Analysis of Training and Prediction Times for REM Construction.

Figure 5 presents two scenarios: (a) depicts the occupied environment, there are four occupants present, and the areas nearest to them exhibit weaker signal strength due to their bodies acting as signal barriers. Conversely, (b) shows the unoccupied environment. In figure 4(b) displays a smooth signal strength distribution, clearly visualizing that there are no occupants or, in other words, no obstructions affecting the signal.

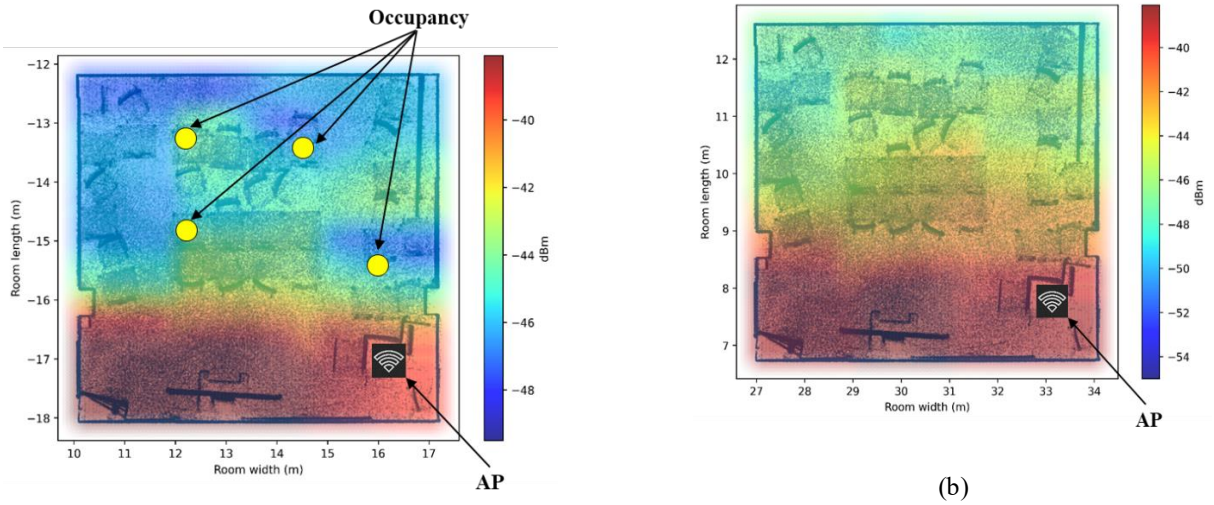


Figure 5. (a) Occupied vs (b) Unoccupied detection using REM.

5.1 Algorithmic Complexity

The computational complexity of the suggested machine learning based ETR method, along with the RFR and BR systems for analogy, is analyzed in this subsection. The results demonstrate that the computational complexity of the ETR depends on factors such as the number of regression trees (T), the number of features (F), the number of samples (S), and the maximum depth of the trees (D_{max}). The complexity can be approximated using $O(T \cdot F \cdot S \cdot D_{max})$, where T represents the number of trees, F is the number of features, S is the number of training samples, and D_{max} is the maximum tree depth. In our simulations, we selected F= 10 and D_{max}=40 for determining the optimal splits [4, 22].

Similarly, the algorithmic complexity of the RFR follows $O(T \cdot F \cdot S \cdot Dmax)$. Nonetheless, the ETR algorithm computes quicker than the RFR because its splitting system is different. It uses a random margin for splitting the data instead of searching for the most optimal threshold. For the BR, the computational intricacy is given by $O(B \cdot C_B)$. B consider as the number of base regressors, and C_B indicates the resource usage for training one base regressor [22].

Conclusion

An alternative method for indoor occupancy detection was studied in this work. REMs were used to predict occupancy and validate the robustness of mapping results. Data were collected from environments with and without occupants. Predictive modeling was employed to assess the accuracy of occupancy detection. Multiple error metrics, including RMSE, were used to evaluate the performance of the proposed system. The Extra Trees Regressor demonstrated superior performance across all error evaluation methods, including RMSE and MAE. Complex variations in indoor environments, such as signal conditions, were effectively handled by this model. The results confirm that the REM-based approach is reliable for accurate occupancy detection. Its potential to optimize energy consumption in smart buildings was also highlighted. Future work aims to enhance the system's scalability by incorporating advanced machine learning techniques. Additionally, real-time feedback will be integrated to enable adaptive energy management in dynamic environments.

References

1. Zou, H.; Zhou, Y.; Yang, J.; Spanos, C.J. Device-free occupancy detection and crowd counting in smart buildings with WiFi-enabled IoT. Energy and Buildings, 174, 309–322 (2018).
2. Salman, M.; Caceres-Najarro, L.A.; Seo, Y.D.; Noh, Y. WiSOM: WiFi-enabled self-adaptive system for monitoring the occupancy in smart buildings. Energy, 294, 130420 (2024).
3. Khan, S.U.; García, C.E.; Hwang, T.; Koo, I. Radio Environment Map Construction Based on Privacy-Centric Federated Learning. IEEE Access, 12, 28109–28121 (2024).
4. Aziz, T.; Camana, M.R.; Garcia, C.E.; Hwang, T.; Koo, I. REM-Based Indoor Localization with an Extra-Trees Regressor. Electronics, 12, 4350 (2023).
5. Aziz, T.; Insoo, K. Enhancing Indoor Localization Accuracy through Multiple Access Point Deployment. Electronics, 13, 3307 (2024).
6. Labinghisa, B.; Park, G.S.; Lee, D.M. Improved indoor localization system based on virtual access points in a Wi-Fi environment by filtering schemes. In Proceedings of the 2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN). IEEE, pp. 1–7 (2017).
7. Di Renzo, M.; Zappone, A.; Debbah, M.; Alouini, M.S.; Yuen, C.; de Rosny, J.; Tretyakov, S. Smart Radio Environments Empowered by Reconfigurable Intelligent Surfaces: How It Works, State of Research, and The Road Ahead. IEEE Journal on Selected Areas in Communications, 38, 2450–2525 (2020).
8. Salman, M.; Caceres-Najarro, L.A.; Seo, Y.D.; Noh, Y. WiSOM: WiFi-enabled self-adaptive system for monitoring the occupancy in smart buildings. Energy, 294, 130420 (2024).
9. Akkaya, K.; Guvenc, I.; Aygun, R.; Pala, N.; Kadri, A. IoT-based occupancy monitoring techniques for energy-efficient smart buildings. In Proceedings of the 2015 IEEE Wireless communications and networking conference workshops (WCNCW). IEEE, pp. 58–63 (2015).
10. Han, Z.; Gao, R.X.; Fan, Z. Occupancy, and indoor environment quality sensing for smart buildings. In Proceedings of the 2012 IEEE international instrumentation and measurement technology conference proceedings. IEEE, pp. 882–887 (2012).
11. Zhao, H.; Qi, Z.; Wang, S.; Vafai, K.; Wang, H.; Chen, H.; Tan, S.X.D. Learning-based occupancy behavior detection for smart buildings. In Proceedings of the 2016 IEEE international symposium on circuits and systems (ISCAS). IEEE, pp. 954–957 (2016).
12. Chi¸tu, C.; Stamatescu, G.; Stamatescu, I.; Sgârciu, V. Wireless system for occupancy modelling and prediction in smart buildings. In Proceedings of the 2017 25th Mediterranean Conference on Control and Automation (MED). IEEE, pp. 1094–1099 (2017).
13. Khan, I.; Zedadra, O.; Guerrieri, A.; Spezzano, G. Occupancy Prediction in IoT-Enabled Smart Buildings: Technologies, Methods, and Future Directions. Sensors, 24, 3276 (2024).
14. Brennan, C.; Taylor, G.W.; Spachos, P. Designing learned CO2-based occupancy estimation in smart buildings. IET Wireless Sensor Systems, 8, 249–255 (2018).
15. Alisic, R.; Molinari, M.; Paré, P.E.; Sandberg, H. Ensuring privacy of occupancy changes in smart buildings. In Proceedings of the 2020 IEEE Conference on Control Technology and Applications (CCTA). IEEE, pp. 871–876 (2020).
16. Nesa, N.; Banerjee, I. IoT-based sensor data fusion for occupancy sensing using Dempster–Shafer evidence theory for smart buildings. IEEE Internet of Things Journal, 4, 1563–1570 (2017).
17. Huang, Q.; Ge, Z.; Lu, C. Occupancy estimation in smart buildings using audio-processing techniques. arXiv preprint arXiv:1602.08507 (2016).
18. Chen, Z.; Jiang, C.; Xie, L. Building occupancy estimation and detection: A review. Energy and Buildings, 169, 260–270 (2018).
19. Dong, B.; Li, Z.; Taha, A.; Gatsis, N. Occupancy-based buildings-to-grid integration framework for smart and connected communities. Applied energy, 219, 123–137 (2018).
20. Luppe, C.; Shabani, A. Towards reliable intelligent occupancy detection for smart building applications. In Proceedings of the 2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE). IEEE, pp. 1–4 (2017).
21. Acosta, M.R.C.; Ahmed, S.; Garcia, C.E.; Koo, I. Extremely randomized trees-based scheme for stealthy cyber-attack detection in smart grid networks. IEEE Access, 8, 19921–19933 (2020).

22. García, C.E.; Koo, I. Extremely Randomized Trees Regressor Scheme for Mobile Network Coverage Prediction and REM Construction. IEEE Access, 11, 65170–65180 (2023).
23. Hwang, T.; Acosta, M.R.C.; Moreta, C.E.G.; Koo, I. Estimating Indoor Radio Environment Maps with Mobile Robots and Machine Learning. Int. J. Adv. Smart Converg., 12, 92–100 (2023).